mark the program unit as waiting for i/o (input/output) so that the program unit continues execution when the service unit returns with data. The information about which service unit was accessed may be stored.

[0053] In phase **625**, the service unit detects that it has been accessed for service and issues an access signal to the processor (or one of the processors). This access signal may be a message (a software signal or an exception) or it may be a state change on a hardware line such as a hardware interrupt line, or a change in the contents of a hardware register/memory.

[0054] In phase **630**, the service unit may send an access signal in response to detecting that the service unit has been accessed. This has been explained in the context of FIG. **3**.

[0055] In phase **635**, the access signal is received at the processor and the access signal is handled e.g. through interrupt processing in phase **640**. The authorization of the program unit to access the service unit may be determined. For example, the program unit may have a certain level of access rights or certain listed rights in its possession. These access rights may be used e.g. by the operating system to control the access of this program unit to service units. This may happen in various ways, for example through determining in phase **645** what kind of action, if any, to take when a program unit accesses a service unit. For example, the operating system may set up or modify the interrupt vector table so that the table contains interrupt vectors (pointers to interrupt handlers) that are specific to the current program unit being executed on the processor. Each program unit may have its own set of interrupt vectors, or the units may have common interrupt vectors. Some or all of the interrupts may be masked so that the interrupt handlers are not called, e.g. so that interrupts coming from service units that the program unit is authorized to access are masked.

[0056] If it is determined in phase **645** that the program unit has access rights to access the service unit, the operation of the program unit, that is the execution of the program unit on the operating system, will continue normally in phase **650**.

[0057] In phase **670**, the program unit that accessed the service unit may be terminated if it was determined in phase **645** that the program unit has no access rights to access the service unit, or that the access rights are insufficient. That is, if the program unit was not authorized to access the service unit, the processor/operating system may kill the program unit. This may happen e.g. so that an interrupt handler is executed in phase **660**, and the handler contains code that will cause the processor and the operating system to terminate the program unit (kill it), put it on hold by marking it not allowed for execution, or reduce its priority. Before doing this, the identity of the program unit to be terminated may be determined in phase **665** by e.g. from process number communicated in the interrupt, or by checking from the operating system which process has accessed the service unit in question.

[0058] In phase **675**, the user may be alerted that an unauthorized access has taken place, and/or the program unit has been terminated.

[0059] The various embodiments of the invention can be implemented with the help of computer program code that resides in a memory and causes the relevant apparatuses to carry out the invention. For example, an apparatus may comprise circuitry and electronics for processing, receiving and transmitting data, computer program code in a memory, and a processor that, when running the computer program code, causes the apparatus to carry out the features of an embodiment. Yet further, a module may comprise circuitry and electronics for processing, receiving and transmitting data, computer program code in a memory, and a processor that, when running the computer program code, causes the module to carry out the features of an embodiment. The various embodiments may be implemented as a computer program product that is suitable for running on the apparatus or the module. The computer program product may be embodied on a computer-readable medium such as a non-transitory permanent storage medium, a memory, or as a signal.

[0060] It is obvious that the present invention is not limited solely to the above-presented embodiments, but it can be modified within the scope of the appended claims.

1-57. (canceled)

58. A method, comprising:

accessing from a program unit a service unit for service,

receiving an access signal related to said service unit in response to said accessing,

determining whether said accessing is authorized, and

if said accessing is not authorized, terminating said program unit.

59. A method according to claim **58**, wherein said service unit is a hardware unit and said access signal is a hardware signal such as a hardware interrupt from said service unit.

60. A method according to claim **58**, wherein said signal is a software interrupt or a software exception from said service unit.

61. A method according to claim **58**, wherein said signal comprises information indicative of said program unit.

62. A method according to claim **58**, wherein said signal is an interrupt and said method comprises:

setting up an interrupt handler for handling said interrupt,

receiving said interrupt,

handling said interrupt in said interrupt handler, and

terminating said program unit with said interrupt handler.

63. A method according to claim **62**, comprising:

setting up said interrupt handler in response to said program unit not having rights to access said service unit, and

masking said interrupt in response to said program unit having rights to access said service unit.

64. A method according to claim **58**, wherein said accessing is authorized if said program unit has rights to access said service unit.

65. An apparatus comprising at least one processor, at least one memory including computer program code for one or more program units, the at least one memory and the computer program code configured to, with the processor, cause the apparatus to perform at least the following:

access from a program unit a service unit for service,

receive an access signal related to said service unit in response to said accessing,

determine whether said accessing is authorized, and

if said access is not authorized, terminate said program unit.

66. An apparatus according to claim **65**, further comprising a hardware signal line for receiving said access signal from said service unit, wherein said service unit is a hardware unit and said access signal is a hardware signal such as a hardware interrupt.

67. An apparatus according to claim **65**, wherein said signal is a software interrupt or a software exception from said service unit, and said apparatus further comprising computer